

Republic of China Patent Certificate  
中華民國專利證書

Invention # 138266  
發明第一三八二六六號

Name of Invention: Table Format Programming  
發明名稱：表格格式程式設計

Owner of Patent: Grow With Me, Inc.  
專利權人：與我成長有限公司

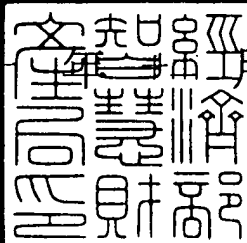
Inventor: Peter A. Lam  
發明人：林亞夫

Patent Effective Date: 7/21/2001 - 10/6/2019  
專利權期間：自中華民國九十年七月二十一日  
至一〇八年十月六日止

The above owner of patent is awarded patent right according to patent law.  
上開發明業經專利權人依專利法之規定取得專利權

Dept. of Finance, Commission of Intellectual Properties  
經濟部智慧財產局  
Commissioner 局長 陳明邦

中華民國



三 日

Page Blank (uspto)

# 中華民國專利公報 [19] [12]

[11]公告編號：446914

[44]中華民國 90年 (2001) 07月 21日

發明

全 16 頁

[51] Int.Cl 06: G06F9/45

Name	[54]名 稱：表格格式程式設計	Date of Application
App #	[21]申請案號：088117272	[22]申請日期：中華民國 88年 (1999) 10月 07日
Priority	[30]優先權：[31]09/169,462	[32]1998/10/09 [33]美國 USA
Inventor	[72]發明人：林亞夫	美國 USA
Applicant	[71]申請人：與我成長有限公司	美國 USA
Attorney	[74]代理人：彭秀霞 先生	

## claims

[57]申請專利範圍：

(Corresponds to claim 89)

1. 一種由計算設備執行的電腦程式，包括具有 x 種配置狀態的第一表格，所述配置狀態中的至少一種定義了一個或多個限定條件；

規定了 y 個路徑的第二表格，當滿足所述第一表格中列出的一個限定條件時，執行所述路徑中的至少一個；所述程式還包括下述表格中的至少一個：

(1) 一個規定用於表示一種程式設計語言的相應預定義指令的用戶定制運算式的表格；

(2) 一個定義關鍵字的可取字格式的表格；

(3) 一個用 m 種任務狀態定義 n 個任務約有效性的表格，所述任務狀態中的至少一種規定了 k 個有效的選定任務；

(4) 一個用 p 種任務狀態定義 q 個任務的表格，所述任務狀態中的至少一個規

定了將被服務約有效任務的優先權；

(5) 一個定義 x 個限定運算式的表格，每個限定運算式表示一種配置狀態的限定條件；以及

(6) 一個定義 y 個輸出運算式的表格，每個輸出運算式表示一種配置狀態或所述第二表格中列出的一個路徑元素的輸出條件。

(claim 90)

2. 如申請專利範圍第 1 項所述之電腦程式，其中用存儲在用於存儲數位資料的存儲裝置中資料來表示。

(claim 91)

3. 如申請專利範圍第 2 項所述之電腦程式，其中該程式被嵌入用於銷售商品的計算設備中。

4. 一種用於對回應一個或多個限定條件執行一個或多個路徑的計算設備進行程式設計的程式設計方法；所述程式設計方法最少包括步驟：

(1) 規定 x 種配置狀態；

(2) 規定一個或多個限定條件到至少一

(2)

3

狀態，

規定 y 個由所述計算設備執行的路徑；

(4)分配 z 個標號來表示步驟(3)中的一個或多個路徑元素，其中 z 是大於等於 1 的整數；

(5)對於步驟(2)中的每個限定條件，當滿足一個特定限定條件時，就規定步驟(3)中的一個將被計算設備執行的路徑；

(6)將配置狀態中的至少一種規定為有效配置狀態；

(7)為步驟(4)中的每個分配標號，定義一個用所述標號表示的可執行程式。

5. 如申請專利範圍第 4 項所述之程式設計方法，其中步驟(4)中的標號最初不能被所述計算設備執行並且另一步驟(7)被配置成能夠被所述計算設備執行所述標號。

6. 如申請專利範圍第 5 項所述之程式設計方法，其中還包括識別不能被所述計算設備執行的所述標號以用於構成步驟(7)的程式的步驟。

7. 如申請專利範圍第 4 項所述之程式設計方法，其中用包括表格格式程式設計語言在內的任何可用程式設計語言來構成步驟(7)的程式。

8. 如申請專利範圍第 7 項所述之程式設計方法，還包括定義用於在步驟(1)到(6)所表示的程式和標號所表示的可執行程式之間傳遞參數的裝置的步驟。

9. 如申請專利範圍第 4 項所述之程式設計方法，還包括規定指向於所述 x 種配置狀態中的至少一種的一個或多個輸出條件的步驟。

10. 如申請專利範圍第 4 項所述之程式設計方法，其中在一個路徑或一種配置狀態中規定了至少一個輸出條件。

11. 如申請專利範圍第 4 項所述之程式設計方法，還包括識別所述計算設備管

4

理其狀態和路徑所需的資源的步驟。

12. 如申請專利範圍第 4 項所述之程式設計方法，還包括指示所述計算設備用於構成步驟(7)的程式的資源的步驟。

13. 如申請專利範圍第 4 項所述之程式設計方法，其中所述計算設備由兩個或多個處理器構成；利用第一處理器提供管理特定狀態和路徑的資源並利用第二處理器資源運行步驟(7)的程式的至少一部分。

14. 如申請專利範圍第 4 項所述之程式設計方法，其中所述計算設備被定義為第一計算設備；所述程式設計方法還包括利用通信連接從與所述第一計算設備距離遙遠的第二計算設備接收表示上述步驟的數位資料的步驟。

15. 如申請專利範圍第 4 項所述之程式設計方法，其中至少部分所述步驟按照表格格式進行組織。

16. 如申請專利範圍第 15 項所述之程式設計方法，還包括將步驟(1)和(2)的配置狀態分成一個或多個表格的步驟。

17. 如申請專利範圍第 15 項所述之程式設計方法，其中配置狀態或路徑不必按照順序彼此依次列出。

18. 如申請專利範圍第 4 項所述之程式設計方法，還包括將配置狀態和路徑說明轉換成存在所述計算設備中的數位資料以使用於執行的步驟。

19. 如申請專利範圍第 4 項所述之程式設計方法，還包括用於轉換利用不同格式的第二程式設計語言的所述步驟的至少部分說明的步驟。

20. 如申請專利範圍第 4 項所述之程式設計方法，還包括用於識別由所述步驟構成的程式位置的步驟。

21. 如申請專利範圍第 7 項所述之程式設計方法，其中步驟(7)的程式是一個位於由申請專利範圍 1 中的步驟構成的程式之外的程式。

22. 如申請專利範圍第 21 項所述之程式設計方法，還包括用於識別所述外部程式位置的步驟。
23. 如申請專利範圍第 4 項所述之程式設計方法，還包括用於形成一個獨立的用於規定由步驟(1)到(7)構成的程式是有效還是無效的表格的步驟。
24. 如申請專利範圍第 4 項所述之程式設計方法，還包括用於分配一個表示一種配置狀態的標號的步驟。
25. 如申請專利範圍第 4 項所述之程式設計方法，還包括用於分配一個表示一個路徑的標號的步驟。
26. 一種用於對執行多個程式的計算設備進行程式設計的程式設計方法，包括步驟：
- (1) 規定  $n$  個可由所述計算設備執行的程式；
- (2) 定義  $m$  種任務狀態，每種任務狀態規定  $k$  個有效的選定程式，其中  $k$  是大於等於 0 的整數；以及
- (3) 規定步驟(2)中的一種任務狀態為有效任務狀態。
27. 如申請專利範圍第 26 項所述之程式設計方法，其中至少一個程式是一個表格格式程式，包括：
- $x$  種程式配置狀態，所述配置狀態中的至少一種定義了一個或多個限定條件；
- $y$  個由所述計算設備執行的路徑；以及
- 當滿足一個特定的限定條件時，所述計算設備執行所述路徑中的一個。
28. 如申請專利範圍第 26 項所述之程式設計方法，其中步驟(1)的程式是用不同的語言編寫而成。
29. 如申請專利範圍第 26 項所述之程式設計方法，還包括用於提供一個關鍵字以表示構成步驟(1)到(3)的一個表格格式程式組的步驟。
30. 如申請專利範圍第 26 項所述之程式設計方法，其中所述任務狀態不順序彼此依次列出。
31. 如申請專利範圍第 26 項所述之程式設計方法，其中步驟(1)到(3)定義第一任務表，所述程式設計方法還包括定義一個不同的第二任務表的步驟。
32. 一種用於對一個遠端計算設備程式設計的方法，包括步驟：
- (1) 規定  $x$  種配置狀態，所述配置狀態中的至少一種定義一個或多個限定條件；
- (2) 規定  $y$  個由所述計算設備執行的路徑；
- (3) 對步驟(1)的每個限定條件，當滿足一個特定的限定條件時，規定一個由所述計算設備執行的路徑；
- (4) 規定一種限定配置狀態變成有效配置狀態；
- (5) 將表示上述步驟的數位資料存入一個本地計算設備；以及
- (6) 利用通信連接將步驟(5)的數位資料下載到所述遠端計算設備中。
33. 如申請專利範圍第 32 項所述之方法，其中所述通信連接是為網路。
34. 如申請專利範圍第 33 項所述之方法，其中所述網路包括因網際網路、內聯網、外聯網或 LAN。
35. 如申請專利範圍第 32 項所述之方法，還包括將一個路徑元素指向於以不同格式的第二程式設計語言編寫而成的程式的步驟。
36. 如申請專利範圍第 32 項所述之方法，還包括評估所述遠端計算設備結構的步驟和將上述步驟配置成同所述遠端計算設備結構運行的步驟。
37. 如申請專利範圍第 32 項所述之方法，還包括將所述步驟中規定的至少部分資料形成一個表格格式的步驟。
38. 如申請專利範圍第 32 項所述之方法，還包括將表示所述配置狀態和路徑的

(corresponds to claim 120)

Corresponds to (claim 114)

(4)

7

資料存入所述遠端計算設備中以執行的步驟。

如申請專利範圍第32項所述之方法，還包括將配置狀態和路徑的至少部分說明轉換成不同格式的第二程式設計語言的步驟。

40. 如申請專利範圍第32項所述之方法，其中配置狀態或路徑不必按照順序依次列出。

(Corresponds to claim 124)  
41. 一種由第一處理器和第二處理器構成的多處理器計算設備，其中所述第一處理器用於執行一個具有 m 種配置狀態和 n 個路徑的表格格式程式的至少一部分。

42. 如申請專利範圍第41項所述之多處理器計算設備，其中第二處理器執行用不是表格格式的第二語言編寫而成的程式。

43. 如申請專利範圍第42項所述之多處理器計算設備，其中第二處理器用於執行一個由所述第一處理器執行的表格格式程式所指導的程式。

44. 如申請專利範圍第41項所述之多處理器計算設備，其中所述第一和第二處理器位於積體電路上。

45. 如申請專利範圍第41項所述之多處理器計算設備，其中處理器之一可執行的至少一個指令不同於另一個處理器可執行的指令集。

46. 如申請專利範圍第41項所述之多處理器計算設備，還包括用於在所述第一和第二處理器之間傳遞參數的裝置。

(Corresponds to claim 135)  
47. 一種用於構成一個適於對具有 m 種配置狀態和 n 個路徑的表格格式程式進行編譯的編譯器的方法，包括步驟：  
(1) 識別表示配置狀態的區域；  
(2) 識別表示路徑的區域；  
(3) 識別一種配置狀態的至少一個限定條件並將其與具體的路徑相連接；以及

8

(4) 將配置狀態 A 與一個將引用所述配置狀態 A 作為其元素的路徑相連接。

48. 如申請專利範圍第47項所述之方法，還包括用於將步驟(1)到(4)的功能綜合為一個現有語言的編譯器的過程。

49. 如申請專利範圍第47項所述之方法，在步驟(1)中還包括一個識別表示配置狀態開始部分的關鍵字的過程。

50. 如申請專利範圍第47項所述之方法，在步驟(2)中還包括一個識別表示路徑開始部分的關鍵字的過程。

51. 如申請專利範圍第47項所述之方法，還包括使一個用戶定制運算式與表格格式程式設計語言的一個具體指令相等的步驟。

52. 如申請專利範圍第51項所述之方法，還包括識別一個使用戶定制運算式與表格格式程式設計語言的預定義指令集相等的表格的步驟。

53. 如申請專利範圍第47項所述之方法，還包括根據所述表格格式程式的指令集識別不可執行的標號的步驟。

54. 如申請專利範圍第53項所述之方法，還包括將所述不可執行的標號與一個外部程式相連接的步驟。

55. 如申請專利範圍第54項所述之方法，其中所述外部程式是用包括表格格式程式設計語言在內的任何可用程式設計語言編寫而成的程式組成。

56. 如申請專利範圍第47項所述之方法，還包括區分兩個或多個程式組的步驟，其中每個程式組由至少一個配置狀態表和一个路徑表組成。

57. 如申請專利範圍第56項所述之方法，還包括對所述多個程式組之間的交互作用進行編譯的步驟。

58. 如申請專利範圍第57項所述之方法，還包括定義一個用於啟動一個程式組的指令的步驟。

59. 如申請專利範圍第58項所述之方法，

其中所述指令用一個或多個任務表表示。

(Corresponds to claim 148)

60. 一種對計算裝置程式設計之方法，包括步驟：

- (1) 選擇一種具有預先定義的指令集的程式設計語言；
- (2) 定義用於表示選定語言的一個具體指令的替換運算式；
- (3) 用所述替換運算式編寫程式；以及
- (4) 將所述程式配置成所述計算裝置可執行的程式。

61. 如申請專利範圍第 60 項所述之方法，其中選定的程式設計語言是表格格式程式設計語言，包括：

- x 種配置狀態，所述配置狀態中的至少一種定義了一個或多個限定條件；
- y 個能被所述計算設備執行的路徑；以及
- 當滿足一個特定限定條件時，所述計算設備就執行一個路徑。

62. 如申請專利範圍第 60 項所述之方法，其中步驟(4)是一個由編輯器、編譯器、譯碼器或一個翻譯程式執行的翻譯過程。

63. 如申請專利範圍第 62 項所述之方法，還包括顯示用經過翻譯的選定語言的預定的指令集組成的程式的步驟。

64. 如申請專利範圍第 62 項所述之方法，還包括顯示用用戶定制運算式組成的程式的步驟。

65. 如申請專利範圍第 60 項所述之方法，還包括提供一個位於程式內部的表格以便將所述替換運算式與相應的特定指令進行鏈結的步驟。

(Approx. corresponds to claim 152)

66. 一種用於對回應一個或多個虛擬限定執行一個或多個路徑的計算設備進行程式設計之程式設計方法，所述程式設計方法包括步驟：

- (1) 規定 x 種配置狀態，所述配置狀態中的至少一種定義了一個或多個限定

條件；

(2) 定義步驟(1)中的限定條件中的一個以表示一個虛擬限定；

(3) 規定 y 個由所述計算設備執行的路徑；

(4) 對於步驟(1)中的每個限定條件，當滿足所述限定中的一個具體限定條件時，再規定一個由所述計算設備執行的路徑；

(5) 規定限定配置狀態中的一個變成有效配置狀態。

67. 如申請專利範圍第 66 項所述之程式設計方法，其中所述計算設備被定義為第一計算設備；所述程式設計方法還包括利用通信連接從與所述第一計算設備距離遙遠的第二計算設備接收表示上述步驟的數位資料的步驟。

68. 如申請專利範圍第 66 項所述之程式設計方法，還包括規定一種或多種配置狀態以構成一種輸出配置的步驟。

69. 如申請專利範圍第 68 項所述之程式設計方法，其中所述輸出配置定義計算設備的一個或多個輸出終端的輸出條件。

70. 如申請專利範圍第 68 項所述之程式設計方法，其中所述輸出配置定義由計算設備產生的虛擬計算輸出。

71. 如申請專利範圍第 66 項所述之程式設計方法，還包括將配置狀態和路徑說明轉換成所述計算設備可執行的數位資料的步驟。

72. 如申請專利範圍第 66 項所述之程式設計方法，還包括用於轉換利用不同格式的第二程式設計語言的配置狀態和路徑的至少部分說明的步驟。

73. 如申請專利範圍第 66 項所述之程式設計方法，其中 x 種配置狀態和 y 個路徑不必按照順序彼此依次列出。

74. 如申請專利範圍第 66 項所述之程式設計方法，還包括提供一個用於識別由

(6)

11

步驟組成的程式位置的關鍵字的

如申請專利範圍第66項所述之程式設計方法，還包括將步驟(1)中的配置狀態分成一個或多個表格的步驟，每個表格中有一種配置狀態被規定為有效。

76.如申請專利範圍第66項所述之程式設計方法，還包括構成一個獨立的用於確定由步驟(1)到(7)組成的程式是有效還是無效的表格的步驟。

77.如申請專利範圍第66項所述之程式設計方法，還包括下列步驟：

(6)提供一個預先定義的指令集以便對路徑和配置狀態程式設計；

(7)定義用於表示步驟(6)中的指令集的一個特定指令的替換運算式；

(8)用替換運算式編寫程式；以及

(9)將所述程式配置成所述計算裝置可執行的程式。

78.如申請專利範圍第77項所述之程式設計方法，其中步驟(9)是由編輯器、編譯器、譯碼器或一個翻譯程式執行的翻譯過程。

79.如申請專利範圍第77項所述之程式設計方法，還包括提供一種於替換運算程式與對應特定指令間相互參照之表格的步驟。

80.如申請專利範圍第66項所述之方法，還包括規定一種配置狀態為有效的步驟。

81.如申請專利範圍第66項所述之方法，其中將一個路徑定義為在初始化時將被執行的預設路徑。

82.一種用於在計算設備的程式列表中識別預定類型的字(wordings)之程式設計方法，包括從多種預定字格式(wording style)選擇中選擇一種字格式的步驟。

83.如申請專利範圍第82項所述之程式設計方法，還包括步驟：

12

(1)定義  $x$  種配置狀態，所述配置狀態中的至少一種定義了一個或多個限定條件；

(2)定義由計算設備執行的  $y$  個路徑；以及

(3)當滿足一個具體限定條件時，向計算設備分配一個路徑以執行。

84.如申請專利範圍第82項所述之程式設計方法，其中利用一個具有關鍵字的表格來定義所述字格式。

85.如申請專利範圍第82項所述之程式設計方法，其中所述預定字型被配置成高亮度的用戶定制標號。

86.如申請專利範圍第82項所述之程式設計方法，還包括形成兩組或多組用戶定制字格式並且其中一組被定義為有效的步驟。

87.一種用於對回應一個或多個限定條件執行一個或多個路徑的計算設備進行程式設計之程式設計方法；所述程式設計方法包括步驟：

(1)視定  $x$  種配置狀態，其中  $x$  是大於等於1的整數；

(2)規定一個或多個到於所述  $x$  種配置狀態中的至少一種的限定標號；

(3)使步驟(2)中的限定標號指向一個用於描述限定條件或所述標注所表示的條件的獨立的運算式；以及

(4)規定  $y$  個可執行路徑，其中回應步驟(3)中規定限定條件來執行至少一個路徑。

88.一種計算裝置，包括：

用於執行程式的計算裝置；

用於訪問於一個遠端計算設備相連接的通信連接的裝置；以及

用於存儲所述計算裝置或所述遠端計算設備所執行的數位資料的存儲裝置；其中所述數位資料包括具有  $x$  種配置狀態和  $y$  個路徑的表格格式程式表示；所述配置狀態中的至少一種定

(Approx.  
Corresponds  
to claim  
(75))

(Corresponds  
to claim  
(76))



義了一個或多個限定條件；而在滿足一個提單限定條件時，所述計算裝置執行所述路徑中的一個。

(Corresponds to claim 86)

89. 一種為計算設備進行程式設計之程式設計工具，其包含：

具有第一多數配置狀態的第一表格(113)，所述配置狀態中的至少一種定義了一個或多個限定條件；

規定了第二多數路徑的第二表格，當滿足所述第一表格中列出的一個限定條件時，執行所述路徑中的至少一個；及規定為有效狀態之一種配置狀態；上述工具之特色更包含下述元件及特性之至少一個：

(1)至少一個配置狀態包含虛擬限定；  
(2)至少一個第二多數路徑具有一個或多個標號，其中每個標號代表一個執行程式；

(3)一個或多個表格(103，200)規定使用者定義運算式以代表對應預先定義指令之程式語言；

(4)所述計算設備(803)是經由一通信連接(802)而與本地計算設備(801)連接之遠程電腦；其中數位資料是代表儲存於遠程計算設備(803)中之第一及第二表格經由通信連接而下載至本地計算設備(801)；

(5)所述計算設備(803)包含至少一個第一處理器及一個第二處理器；其中第一處理器是翻譯第一及第二表格關係，同時給予第二處理器執行其他程式之指向；

(6)至少一個配置狀態包含有一個表格及所述表格相等於該表格所代表獨立運算式所定義之限定條件；

(7)表格(104)定義一個較佳形態之關鍵字；

(8)表格(105)具有第三多數之任務狀態以定義第四多數任務之有效；

(9)表格(114)定義輸出運算式之第五多

數，每個輸出運算式代表配置輸出狀況或列於第二表格(116)中，路徑元件；

(10)一個更進一步狀態之表格定義配置狀態之第六多數及一個更進一步路徑表格定義路徑之第七多數；其中第一及第二表格如第一表格組執行第一功能般被組合；及更進一步之狀態路徑表格如第二表格組執行不同第二功能般被組合。

圖式簡單說明：

第一圖表示表格格式程式的一個實施例的結構。

第二圖 A 是用於定義等效於用戶定義運算式的表格格式運算式的表格。

第二圖 B 表示如何將一個名稱分配給第二圖 A 所示的表格。

第三圖是用於定義將被識別的字的列印格式的表格。

第四圖是表示一個任務表的不同任務狀態的實施例。

第五圖 A 是表示一個程式中的多個任務表的實施例。

第五圖 B 是第五圖 A 的數位化表示。

第六圖 A 是表示一個表格格式組的一個實施例的程式。

第六圖 B 是第六圖 A 的實施例的另一種開發形態。

第七圖是表示所包括的文件列表的表格。

第八圖是第七圖所示的表格的另一種形態。

第九圖表示用於表格格式程式設計中的一些建議長度和描述性的指令命令。

第十圖表示現有技術中用於聲音發生微控制器中的基本表格格式程式。

第十一圖是表示一個本地電腦利用一種通信連接將表格格式程式下載到遠端電腦中的方框圖。

- 101 Program: (程序名)
- 106 Include: (定義所包含之程式)
- 107 Define: (定義程式中所用之常數)
- 102 Declare: (變數說明)
- 103 Custom Expression: (用戶自定義之等效命令及語法運算式)
- 104 Identification Style: (用戶自定義之標識類型)
- 105 Task: (一個或多個有效任務表)
- 111 Group: (第一程序組, 通常做為主組之程序組名稱)
- 112 Qualifier: (定義狀態觸發之限定條件)
- 113 State: (觸發狀態表名稱)  
(用於定義配置狀態之表格)
- 114 Output Equation: (定義輸出狀態方程式)
- 115 State: (輸出狀態表名稱)  
(用於定義輸出狀態之表格)
- 116 Path: (用於表達將執行事件之表格)
- 121 Library: (通用函數庫)

第一圖 Figure 1

(9)

201 Custom Expression: MySign 210  
202 (&&) = (AND) 203 // 邏輯及功能 204  
205 (||) = (BIT OR) // 位元至位元或功能  
206 (;) = (//) // 備註  
207 (++) = (INC) // 增加  
208 (!=) = (UNEQ) // 不相等  
---  
209 (%) = (REM) // 餘數

A

216  
Path\_DisplayA\_1: 暫存器A (C:&&) 00001111; 顯示: 暫存器A  
Path\_DisplayA\_2: A: ( 暫存器A & 00001111); 顯示: 暫存器A  
217

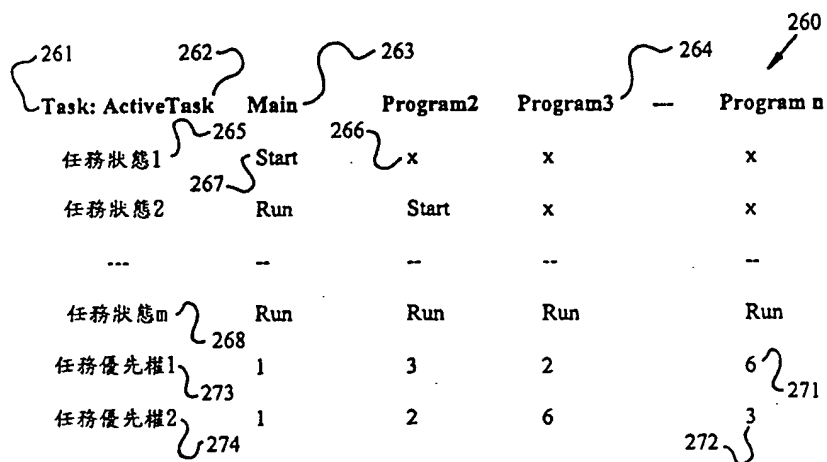
B

第二圖 Figure 2

231 Identification Style: MyStyle 232  
// 此部份定義標號或關鍵字之字體及外形 230  
234 情況 = 標題 235 // 選擇關鍵字之首字母 233  
字體 黑體, 斜體, 底線 // 定義識別字為黑體  
239 240 // 斜體或底線類型

第三圖 Figure 3

(10)



第四圖 Figure 4

Task: Input	Keyboard	Mouse	GamePort
All:	Run	Run	Run
Normal:	Run	Run	x
Keyboard:	Run	x	x
Mouse:	x	Run	x
Game:	x	x	Start
Task: Ports	COM1	COM2	ParallelPort
All:	Run	Run	Run
SrMouse:	Start	x	x
ExModem:	x	Continue	x
Printer:	x	x	Continue
Task: Device	CDRom	HardDriveC	Floppy
ReadCD:	Start	Run	x
HDFullSpeed:	x	Run	x
RWFloppy:	x	Run	Run
ALL:	Run	Run	Run

//可能之任務狀態:開始,繼續,進行,暫停, x

第五圖 A Figure 5A

(11)

301	Task: Input	302 Keyboard	303 Mouse	304 GamePort
305	All:	Run	Run	Run 310
306	Normal:	Run	Run	x
307	Keyboard:	Run	x	x
308	Mouse:	x	Run	x 311
309	Game:	x	x	Start
321	Task: Ports	322 COM1	323 COM2	324 ParallelPort
325	All:	Run	Run	Run
326	SrMouse:	Start	x	x
327	ExModem:	x	Continue	x 329
328	Printer:	x	x	Continue
341	Task: Device	342 CDRom	343 HardDriveC	Floppy
345	ReadCD:	Start	Run	x 344
346	HDFullSpeed:	x	Run	x
347	RWFloppy:	x	Run	Run
348	ALL:	Run	Run	Run
349	可能之任務狀態: 開始, 繼續, 進行, 暫停, x			

第五圖 B Figure 5B

(12)

```
1  Group: Main          //WebSale 主程序組
2  Qualifier:
3  Catalog = Icon(1)
4  Purchase = Icon(2)
5  Service = Icon(3)
6  Home = Icon(4)
7  Quit = Icon(5)

8  State: FirstPage    Catalog    Purchase    Service    Home    Quit
9  Ready:              P_catalog  P_purchase  P_service  Start   Start   Bye
10 Hold1:              x          x          x          Start   Start   Bye

11 State: Response    Group:Info  Group:Order  Group:Service  Group:Register P3.1
12 WindowCatalog:    Run        x          x          x          x
13 WindowPurchase:    x          Run        x          Run        x
14 WindowService:    x          x          Run        Run        x
15 Hold2:            x          x          x          x          x
16 Hold3:            x          Run        x          Run        x
17 Beep:             Continue   Continue   Continue   Continue   P+
```

18 Path:

19 Start:            CheckSystem; DisplayFirstPage; Beep; Hold 2; Ready; END

20 P\_catalog:        WindowCatalog; BuySolicit; END

21 P\_purchase:       WindowPurchase; Hold1; Hold3; GrayButton; END

22 P\_service:        WindowService; BuySolicit; END

23 Bye:             Terminate; END

24 EOG              //表示程序組結束之關鍵字

//注意:行號是為了方便描述,在實際中不需要  
//程式設計:狀態及路徑方程式不必按照順序排列

第六圖 A Figure 6A

(13)

```
1  Group: Main          //WebSale 主程序組

2  Qualifier:
3  Catalog = Icon(1)
4  Purchase = Icon(2)
5  Service = Icon(3)
6  Home = Icon(5)
7  Quit = Icon(4)

8  State: FirstPage    Catalog      Purchase      Service      Home      Quit
9  Ready:              P_catalog    P_purchase    P_service    Start     Bye
10 Hold1:              x          x          x          Start     Bye

11 State: Response     Group:Info    Group:Order    Group:Service Group:Register P3.1
12 WindowCatalog:      Run          x          x          x          x
13 WindowPurchase:     x          Run          x          Run        x
14 WindowService:      x          x          Run          Run        x
15 Hold2:              x          x          x          x          x
16 Hold3:              x          Run          x          Run        x
17 Beep:               Continue     Continue     Continue     Continue   P+

18 Path:
19 Start:              EJ_CheckSystem; EVB_DisplayFirstPage; Beep; Hold 2; Ready; END
20 P_catalog:          WindowCatalog; Lbry:BuySolicit; END
21 P_purchase:         WindowPurchase; Hold1; Hold3; Lbry:GrayButton; END
22 P_service:          WindowService; Lbry:BuySolicit; END
23 Bye:               EJ_Terminate; END

24 Library: Local
25 BuySolicit:         EC_CheckRecord; EVC_SolicitWindow; END
26 GrayButton:         EC_CheckX; EVB_X_Is_Gray; END

27 EOG                //表示程序組結束之關鍵字
```

//注意:行號是為了方便描述,在實際中不需要  
//程式設計:狀態及路徑方程式不必按照順序排列

第六圖 B Figure 6B

(14)

```
Include:
ReadIR           //從遠程控制讀取信號
ReadPanel        //從控制面板讀取信號
ReceiveFromLine  //從電纜線輸入數據
SendToLine       //向電纜線傳輸數據
CompressBuffer   //在向電纜線發送之前壓縮數據
DecompressBuffer //對數據文件進行解壓縮
CompareTime      //比較暫存器時間及實時時鐘
GenIcon          //在電視銀螢幕上生成圖標
GenWindow        //建立一個新視窗
InputCursor      //輸入並譯碼游標之移動
OutputCursor     //更新游標在電視螢幕上之位置
DisplayData      //在電視螢幕上顯示數據
DisplayPicture   //在電視螢幕上顯示圖形文件
.....
.....
ErrorMessage     //對錯誤代碼進行譯碼並傳輸錯誤消息
Help             //顯示幫助螢幕
```

第七圖 Figure 7

```
Include:
EA_ReadIR        //從遠程控制讀取信號
EE_ReadPanel     //從控制面板讀取信號
EA_ReceiveFromLine //從電纜線輸入數據
EA_SendToLine    //向電纜線傳輸數據
EC_CompressBuffer //在向電纜線發送之前壓縮數據
EC-DecompressBuffer //對數據文件進行解壓縮
EA_CompareTime   //比較暫存器時間及實時時鐘
EJ_GenIcon       //在電視銀螢幕上生成圖標
EJ_GenWindow     //建立一個新視窗
EVC_InputCursor  //輸入並譯碼游標之移動
EVB_OutputCursor //更新游標在電視螢幕上之位置
EVB_DisplayData  //在電視螢幕上顯示數據
EVB_DisplayPicture //在電視螢幕上顯示圖形文件
.....
.....
EA_ErrorMessage  //對錯誤代碼進行譯碼並傳輸錯誤消息
EVB_Help         //顯示幫助螢幕
```

第八圖 Figure 8



(15)

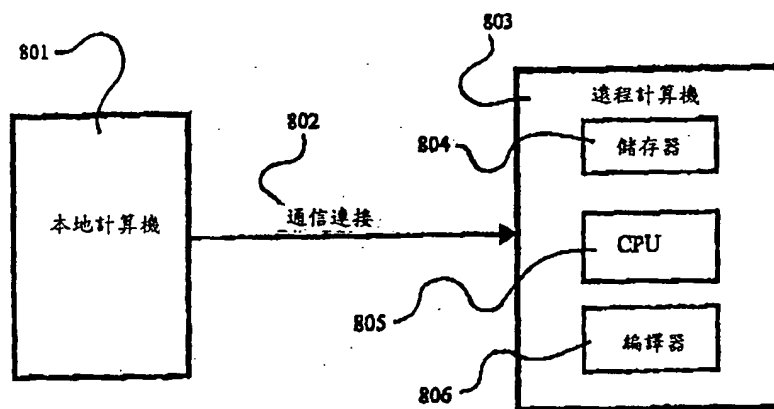
功 能	表格格式指令
<u>指令分離器</u> 標識指令之結束	
<u>指令之繼續</u> 標識後續行	& (在行首)
<u>一元運算器</u> 方向 地址 負的 邏輯否定 1的餘數 增加 減少 大小	VALUE (地址) ADDRESS (值)  NOT (運算式) COMPLEMENT (運算式) INCREMENT DECREMENT SIZEOF (運算式)
<u>移位操作器</u> 左移 右移	BIT SHIFT LEFT (操作數, n) BIT SHIFT RIGHT (操作數, n)
<u>關係操作器</u> 小於 大於 小於或等於 大於或等於	< > <= or == >= or ==
<u>邏輯操作器</u> 位元 AND 邏輯 AND 唯一 OR 位元 OR 邏輯 OR 條件分支	BIT AND AND XOR BIT OR OR 運算式?: [正確語句] [錯誤語句]
<u>注解</u>	// (至文件末端)

第九圖 Figure 9

輸入狀態:						
觸發端子:	TG1	TG2	TG3	TG4	TG5	TG6
State0:	R:Path1	R:Path2	R:Path3	R:Path4	X	X
State1:	F:Path11	R:Path2	R:Path3	R:Path4	X	X
State2:	R:Path1	F:Path11	R:Path3	R:Path4	X	X
State3:	R:Path1	R:Path2	F:Path11	R:Path4	X	X
State4:	R:Path1	R:Path2	R:Path3	F:Path11	X	X
Path:						
Path 1:	State1	Sound1	Path1			
Path 2:	State2	Sound2	Path2			
Path 3:	State3	Sound3	Path3			
Path 4:	State4	Sound4	Path4			
Path11:	State0	結束				

第十圖 Figure 10

(16)



第十一圖 Figure 11